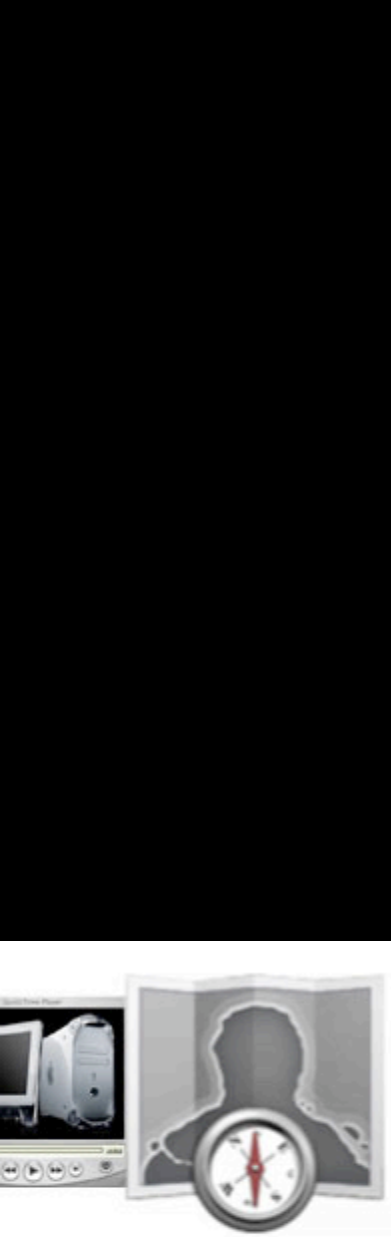


Java Web Services

Session 405

Java Web Services

Stephen Lewallen
Java Classes Technical Lead

What Is a “Web Service”

- An automated application accessible via common protocols (http, smtp, etc.) which can be discovered and used over the network using XML as the mechanism for data exchange

“Free” Web Services Client

- Mac OS X J2SE 1.4
  - Includes building blocks for client
    - XML Parser
    - Network API
  - Fully compatible with Open Source
    - Apache SOAP/AXIS Framework
      - Also leveraged by WebObjects Web Service (client and server)
- Mac OS X J2SE 1.3.1
  - Easy to add needed packages

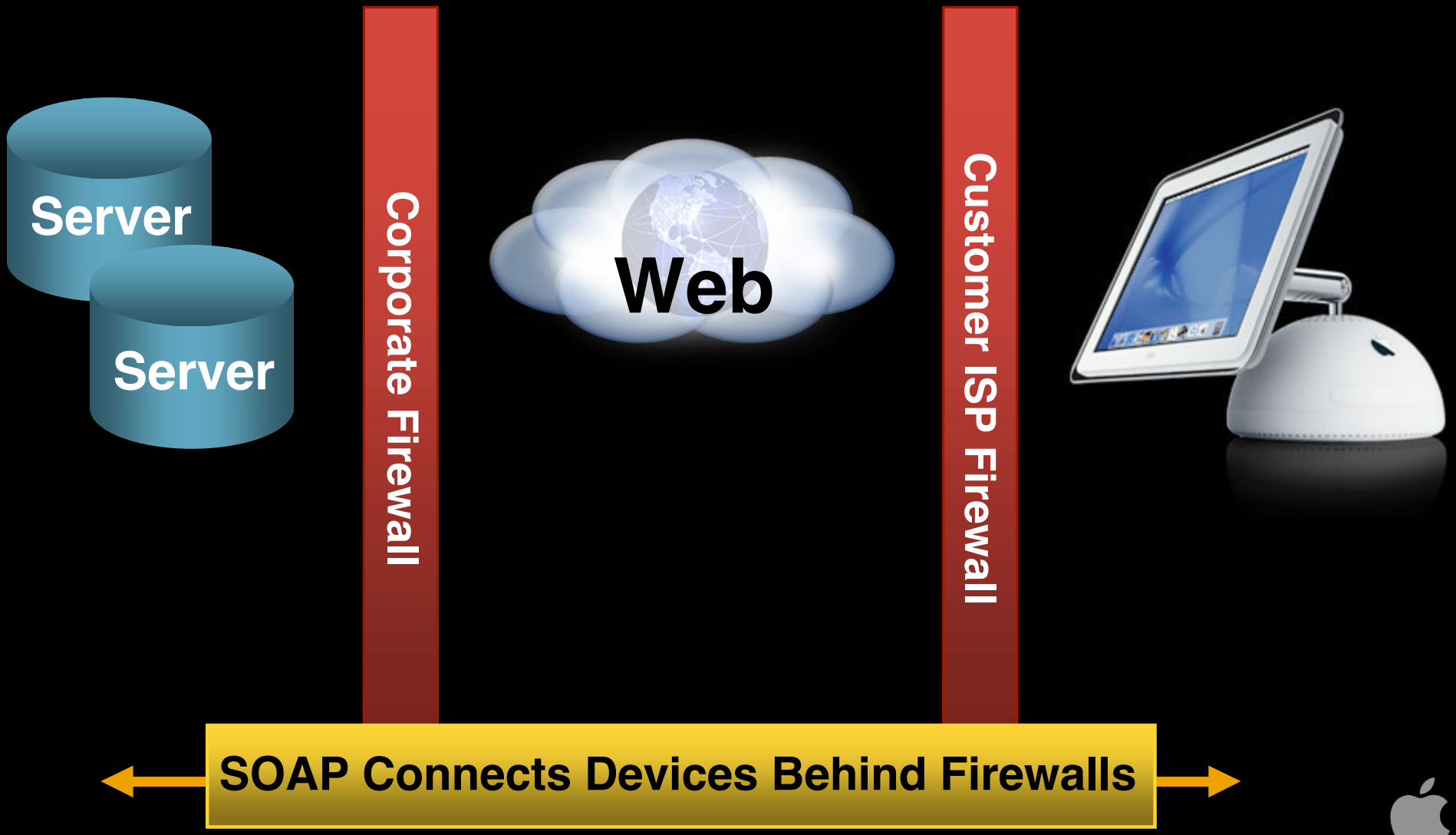
Building Blocks

- Accessing a web service from Java requires:
  - SOAP
    - XML Parser
    - Network stack
      - TCP/IP
      - Protocol support (HTTP, SMTP, etc.)

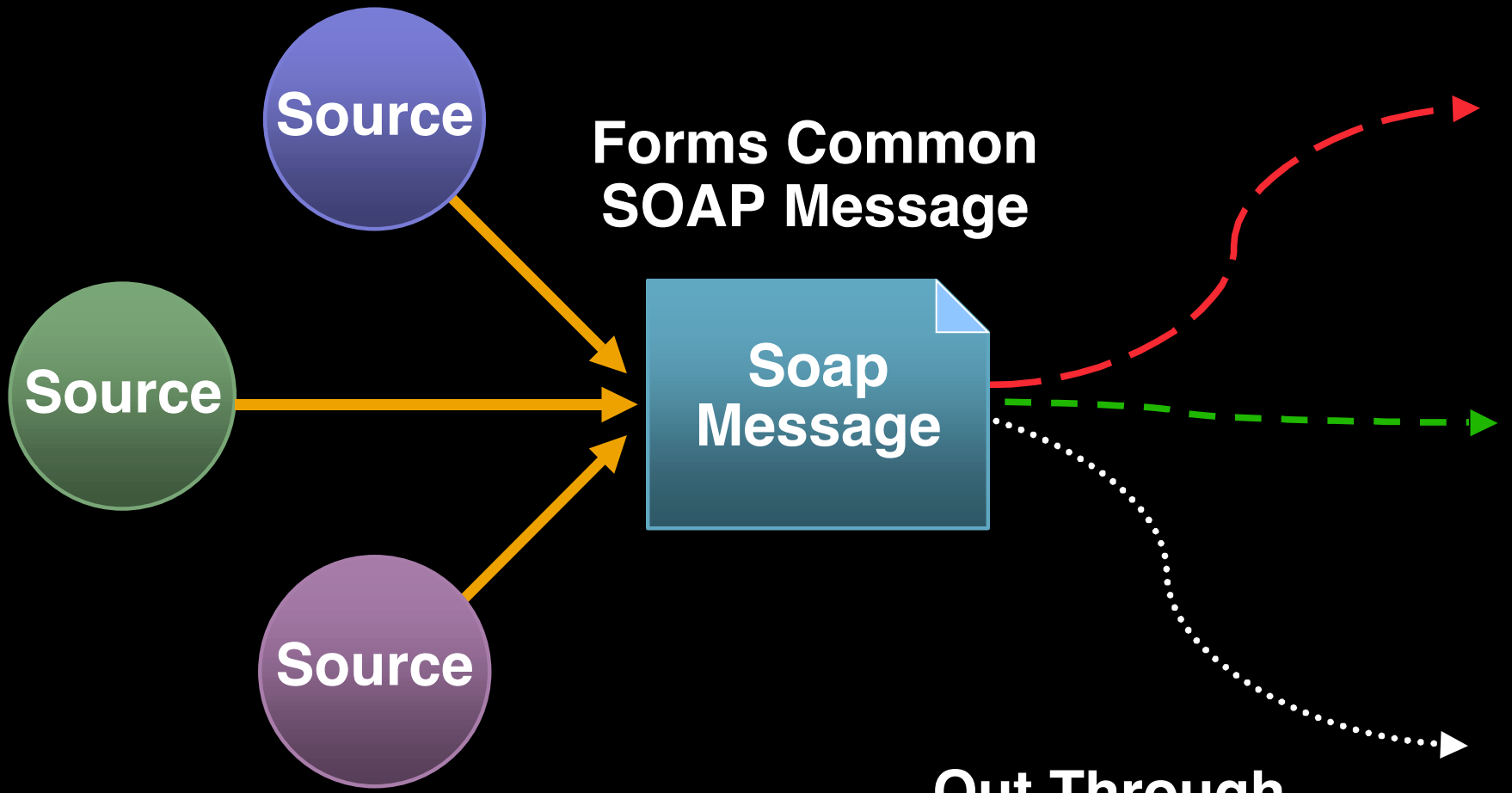
What Is SOAP?

- Simple Object Access Protocol
  - "Lightweight [XML-based] protocol for exchange of information in a decentralized, distributed environment"
    - Created to enable any device or application to communicate with any other device or application over many different protocols

SOAP Means Freedom



SOAP Means Independence



Not Just B2B

- Freedom and independence enable
  - Business to Consumer
    - Ex. “Did you receive shipment?”
  - User to User
    - Ex. Messaging
  - Device to Device

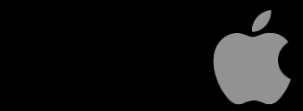
Real World Applications

- UC Berkeley
  - Deploying unified messaging system using SOAP
- UK's "eGovernment Interoperability Framework"
  - Central and local governments will share information using SOAP and UDDI (Universal Description, Discovery, and Integration)

SOAP Technical Overview

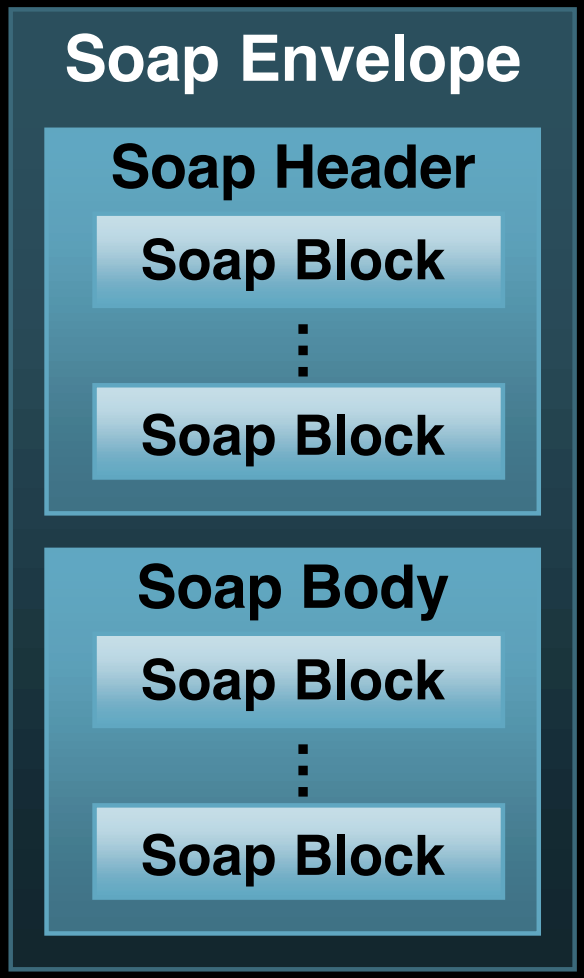
SOAP Overview

- XML-based messaging
- Transport neutral
  - SOAP message defines:
    - Contents of message
    - Who it is intended for
    - Optional attributes
  - SOAP message does not define:
    - How to send or receive SOAP message
  - Inherently “one way” although . . .
    - “Defacto” http(s) transport is two-way request/response
  - Defines fault reporting mechanism



SOAP Format

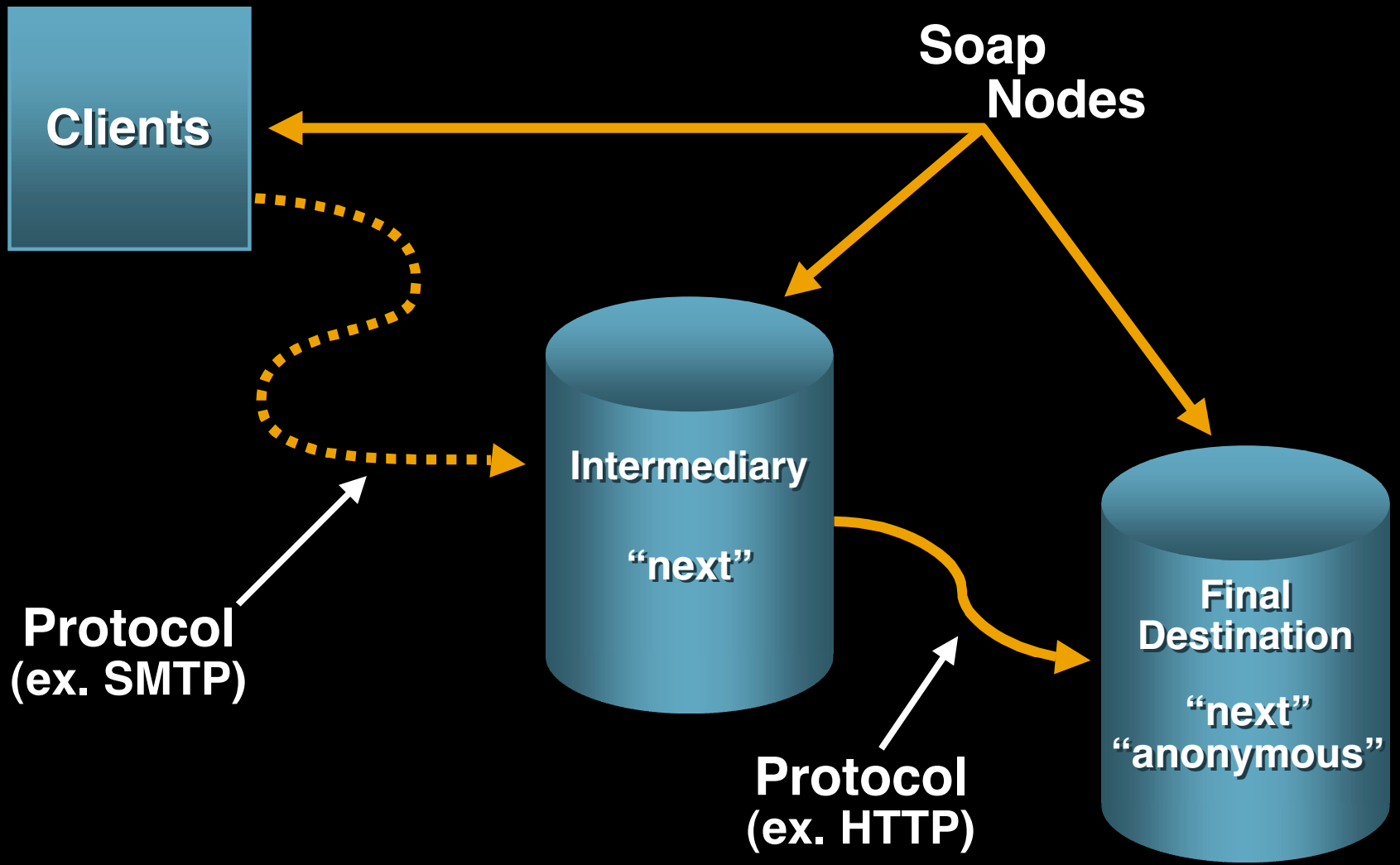
- Envelope: Outermost XML element
  - Header(s): Optional child element of Envelope
    - Used to define additional semantics, such as a transaction protocol
    - Specifies whether receiving must understand header
  - Body: Contains actual message



Nodes, Roles, Actors

- SOAP messages processed by SOAP “Nodes”
  - Nodes play different types of roles
    - SOAP Header “Actor” attribute indicates role(s) node optionally must play
      - “Next”: All nodes must process “next” headers
      - “Anonymous”: Final node must process
      - “None”: Such headers are used in conjunction with other headers
      - Custom: Defined by business message semantics

SOAP Processing Model



Envelope Sample

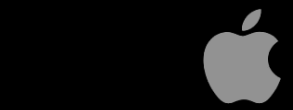
```
1.  <?xml version="1.0" ?>
2.   <env:Envelope xmlns:env="http://www.w3.org/2001/12/soap-envelope">
3.     <env:Header>
4.       <p:oneBlock xmlns:p="http://example.com"
       env:actor="http://example.com/Log">
5.       ::::
6.       </p:oneBlock>
7.       <q:anotherBlock xmlns:q="http://example.com"
8.             env:actor="http://www.w3.org/2001/12/soap-envelope/actor/next">
9.       ::::
10.      </q:anotherBlock>
11.      <r:aThirdBlock xmlns:r="http://example.com">
12.      ::::
13.      </r:aThirdBlock>
14.    </env:Header>
15.    <env:Body >
16.      ::::
17.    </env:Body>
18.  </env:Envelope>
```

SOAP Data Types

- XML Data Types Schema
  - Simple Types: int, float, string
  - Enumerations
  - Byte Array (ex. Base64 encoding)
  - Compound Types: struct, array
    - Varying Array: offset at ‘3’
    - Sparse Array: specific positions ‘1’ and ‘7’

SOAP RPC

- Requires:
  - Valid URI: Binding point for protocol
  - Method name
  - Parameters to method (in, out, or in/out)
- May optionally specify additional headers (ex. Transaction ID)



Example SOAP RPC Message

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <SOAP-ENV:Envelope
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3.                  xmlns:SOAP-
   ENV="http://schemas.xmlsoap.org/soap/envelope/"
4.
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-
   instance">
5.   <SOAP-ENV:Body>
6.     <ns1:echoString xmlns:ns1="http://soapinterop.org/">
7.       <testParam xsi:type="xsd:string">Hello!</testParam>
8.     </ns1:echoString>
9.   </SOAP-ENV:Body>
10. </SOAP-ENV:Envelope>
```

SOAP RPC Reply

- SOAP Struct
  - Return value
  - Out and in/out parameters
- OR Fault
  - Fault code
  - Fault string
  - Fault actor
  - Fault detail

Example SOAP RPC Reply

1. <?xml version="1.0" encoding="UTF-8"?>
2. <SOAP-ENV:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
3. <SOAP-ENV:Body>
4. <ns1:**echoStringResponse** xmlns:ns1="http://soapinterop.org/">
5. <result xsi:type="xsd:string">Hello!</result>
6. </ns1:**echoStringResponse**>
7. </SOAP-ENV:Body>
8. </SOAP-ENV:Envelope>

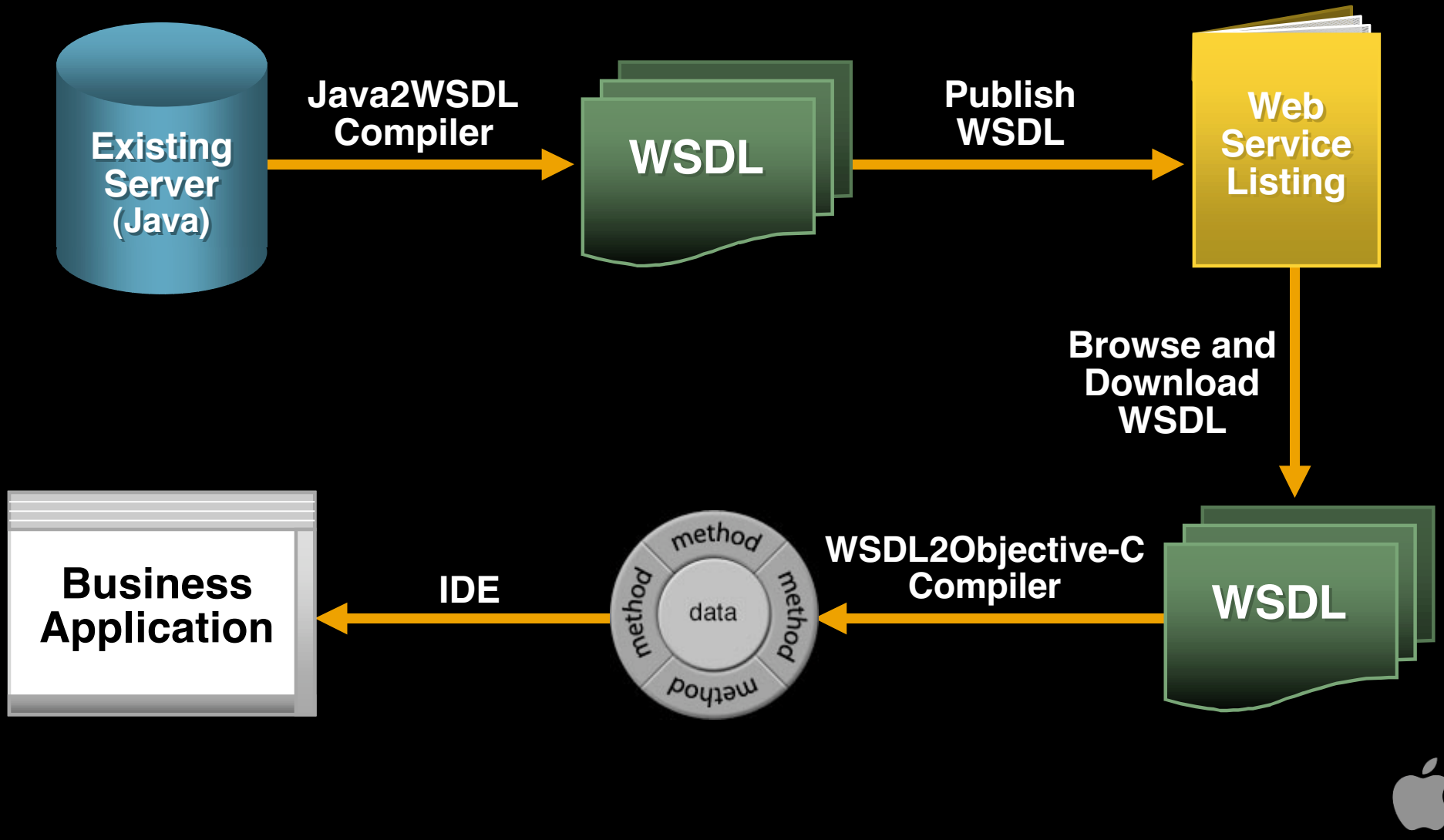
Web Services Description Language

Web Service IDL

WSDL

- Web Services “IDL” in XML Schema
- Specifies
    - Interface to service
        - Method name
        - Parameter types used
    - Location of service (ex. URI)
- Typically used to provide client interface to existing web services
- May also be used to initially define interface
    - Generate Server Skeleton
    - Generate Client Stubs

Typical WSDL Scenario



Service Discovery

Locating Web Services

Discovering Web Services

- Manual
  - Developer finds service from webpage listings
- Assisted
  - Tools assist in discovery of desired service
    - UDDI: Service Directory (B2B focus)
    - WS-Inspection: Web Server Link

Manual Discovery

- The “Yahoo” approach
  - Web services manually cataloged
  - Web services manually discovered
    - Search engine
    - Index
    - Categorization (business, function, feature, etc.)
  - Examples:
    - http://www.salcentral.com
    - http://www.xmethods.com

Semi-Automated Discovery

- Service Lookup well specified
- Integrates Discovery with Tools
    - Provides method and/or API
- Examples:
    - UDDI: Universal Description, Discovery, and Integration
        - Industry initiative for Publish and Discovery
    - WS-Inspection
        - Specifies method on inspecting web site for services
    - DISCO
        - Microsoft (legacy) Publish and Discovery mechanism

Universal Description, Discovery, and Integration

- Web Services Directory Specification and Operational Registry
  - Specification driven by IBM and Microsoft
  - Universal Registry run by Operators (Microsoft, IBM, HP, and SAP)
- Provides web browser and SOAP RPC access for Publish and Discovery
- Private implementations
  - WebObjects 5.2 will include UDDI Intranet Server

UDDI

- Businesses register at any Operator:
  - Business Entity: Name of business, type of business, URL to web site, UUID, etc.
  - Business Services: What services are provided
  - Specification(s): Technical information on how to use services
  - Service Types: Methods to categorize business and services
    - Common industry taxonomies

UDDI Operators

- Each Operator has Registry
- Information is replicated among Operators
    - Not real time (average 24-hour cycle)
    - Registration takes place at any Operator site
    - Searches conducted at any Operator site

UDDI SDKs

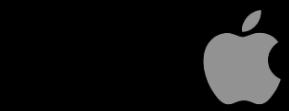
- UDDI4J
    - Java SDK: IBM, HP, and SAP
- Microsoft Visual Studio .NET
- Various Web Service SOAP RPC Servers to Registry
- WebObjects 5.2 includes UDDI v2 Database and EOModel

Demo

SOAP: Mac OS X News Finder

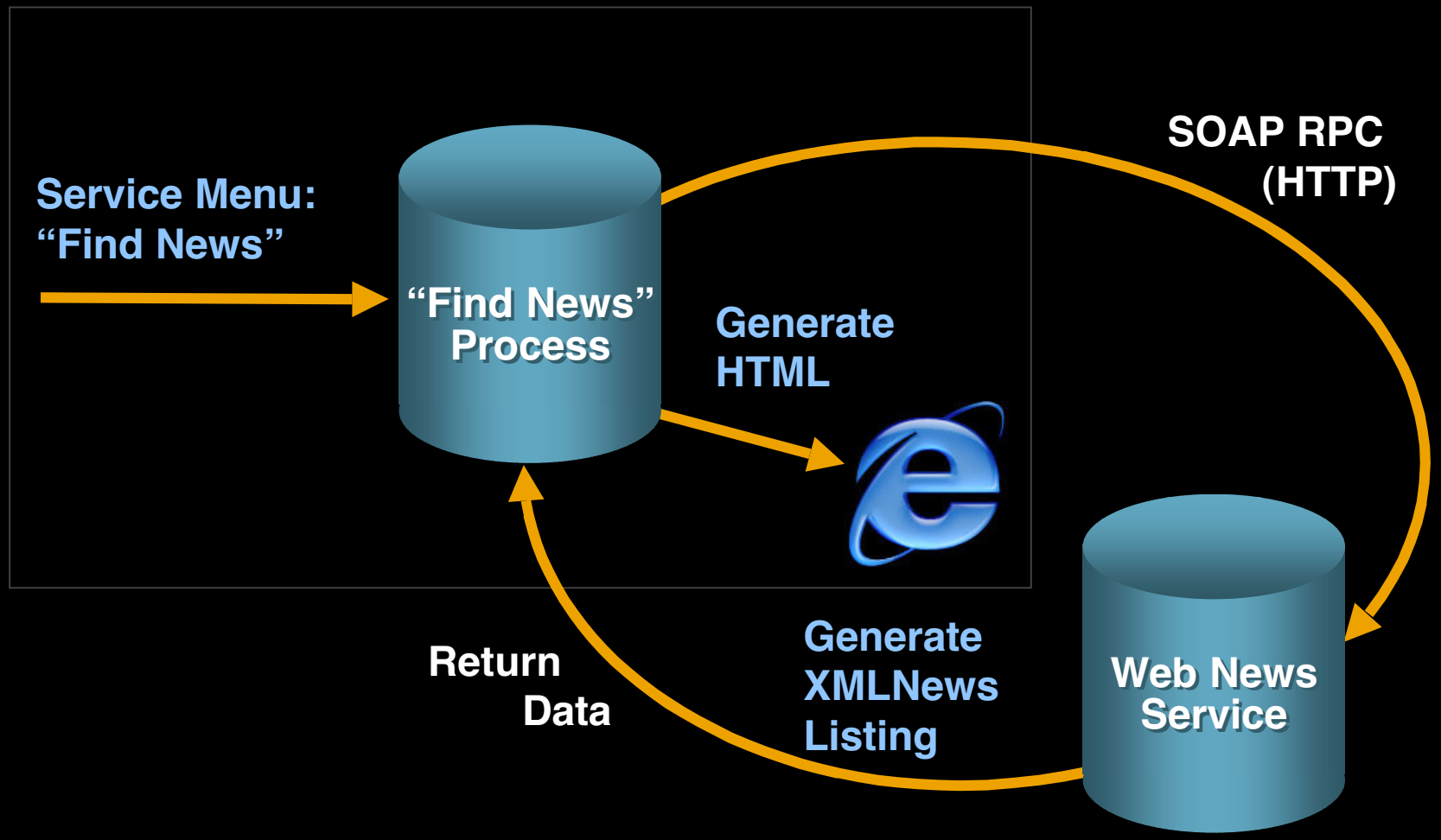
Technologies of Demo

- Java
  - J2SE 1.4
    - Built-in XML Parser
  - Axis SOAP Toolkit
- Objective-C
  - Mac OS X Service (~Library/Services)



Mac OS X News Finder

Mac OS X Client



URL References

- SOAP Web Services
  http://www.w3.org/2002/ws/
- WSDL
  http://www.w3.org/TR/wsdl
- WS-Inspection
  http://www-106.ibm.com/developerworks/library/ws-wsilspec.html
- UDDI
  http://www.uddi.org
- Web Service Listings
  http://www.salcentral.com
  http://www.xmethods.com
- SDKs
  - SOAP Axis: http://xml.apache.org/axis
  - Sun JAX: http://java.sun.com/webservices

How to Access Documentation

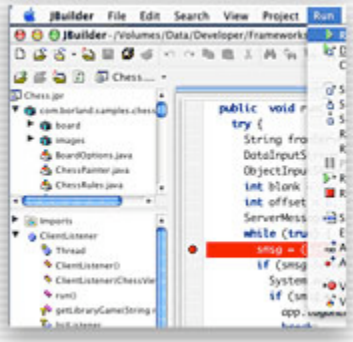
- Most up-to-date: PDF and HTML
  http://developer.apple.com/techpubs/java
- Hardcopy print-on-demand
  Vervante.com under Related Resources
- Product CD
  Documents folder and installed in
  /Developer/Documentation/Java
- Check ADC News for latest updates
  http://developer.apple.com/devnews

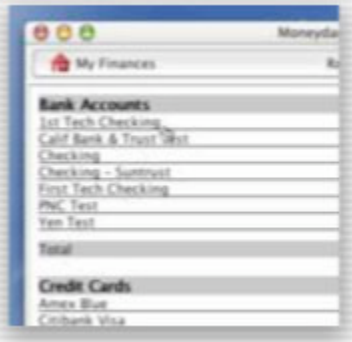
Roadmap

705 WebObjects and Web Services:
WebObjects leverages Java for Web Services

Room A1
Wed., 2:00pm

Q&A





Stephen Lewallen
Java Classes Technical Lead
slewallen@apple.com

http://developer.apple.com/wwdc2002/urls.html


WWDC2002


WWDC2002


WWDC2002